



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/729,228	12/05/2003	David Hawley	16105-002US3 / 2002P00003	9220
32864	7590	06/25/2007	EXAMINER	
FISH & RICHARDSON, P.C. PO BOX 1022 MINNEAPOLIS, MN 55440-1022			RAMPURIA, SATISH	
ART UNIT		PAPER NUMBER		
2191				
MAIL DATE		DELIVERY MODE		
06/25/2007		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)
	10/729,228	HAWLEY ET AL.
	Examiner Satish S. Rampuria	Art Unit 2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 08 May 2007.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-14 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-14 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) Notice of References Cited (PTO-892)
- 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) Information Disclosure Statement(s) (PTO/SB/08)
 Paper No(s)/Mail Date _____.
- 4) Interview Summary (PTO-413)
 Paper No(s)/Mail Date. _____.
- 5) Notice of Informal Patent Application
- 6) Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed on 05/08/2007.
2. Claims amended by the Applicant: 8 and 9
3. Claims 1-14 pending.

Response to Arguments

4. In response to the applicants argument/comments the “provisional” double patenting rejection should continue to be made by the examiner in each application as long as there are conflicting claims in more than one application unless that “provisional” double patenting rejection is the only rejection remaining in at least one of the applications. See MPEP §804.
5. Applicant’s arguments with respect to claims have been considered but they are not persuasive.

In the remarks, the applicant has argued that:

The Examiner contended that certain portions of page 3, page 5, and figure 3 of UIML disclose these aspects of Applicants' claim 1. (Office Action, pages 9-10). This is not correct. Nowhere does UIML disclose or suggest receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device. In contending that UIML does disclose these two limitations, the Examiner cited the same portion of the page 5 section "UIML as a meta-language." In particular, the Examiner cited "UIML document specifies a mapping of those names to a vocabulary specific to a particular target platform." (Office Action, pages 9-10).

This is very different from the objects of different types and the predefined presentation patterns according to the type of device recited in Applicants' claim 1. For example, the UIML sentence that immediately follows the sentence cited by the Examiner reads, "[f]or example, if the target is Java AWT, the vocabulary might consist of the java.awt and java.awt.event class names, such as Frame, Menu, and Button." Even if these class names may correspond to objects, which Applicants do not concede, they certainly are not a predefined presentation pattern as recited in Applicants' claim 1. Neither does figure 3 of UIML disclose or suggest predefined presentation patterns or identification of a presentation pattern from predefined first and second presentation patterns according to the type of the device, in contrast to the Examiner's contention.

Neither is Applicants' claim 1 obvious in view of UIML. UIML does not provide any details concerning how objects are presented in user interfaces of various devices. While UIML mentions that it discloses an XML language that permits description of a user interface in a device-independent manner, (page 1, Abstract), no details are provided concerning how anything actually gets presented on a user interface. Mapping of names between platforms, as UIML discloses, is an abstract notion devoid of explanation or specifics that would permit one skilled in the art to

arrive at the method of Applicants' claim 1.
Accordingly, claim 1 defines subject matter that is patentable over UIML.

Examiner's response:

In response to Applicant's argument UIML disclose an XML language that permits a declarative description of a user interface in a highly device-independent manner. An objective of UIML is to permit a UIML document to be mapped to any type of user interface, from graphical to speech, and even to those not yet invented (see abstract). UIML clearly teaches the limitation receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device, See FIG. 1 which shows displaying data on multiple devices by receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device. Further, UIML does disclose presentation pattern since it's displaying data on multiple devices of it's (device) own natural langugag (see page 5, section UIML as a meta language). Applicant only makes general allegations. Therefore UIML does disclose the limitations as claimed and the rejection is maintained herein.

In the remarks, the applicant has argued that:

Dependent claim 5 depends from claim 1, and thus is patentable over UIML for at least the reasons described above with reference to claim 1. Additionally, claim 5 recites "wherein the presentation pattern is identified according to the size of the screen." The Examiner contended, in rejecting claim 5, that UIML discloses this aspect at figure 1, figure 3, and related discussion. (Office Action, page 11). This is not correct. Figure 1 of UIML simply shows

a block diagram having two devices, and describes one as presenting information in English and the other as presenting information in French. Figure 3 also shows two devices, but does not show or describe differences in presentation, let alone differences in presentation based on screen size, and nowhere in UIML are these aspects disclosed or suggested. As described above, UIML does not disclose or suggest predefined presentation patterns, and certainly does not disclose or suggest identifying a presentation pattern based on size of the screen of the device. Without limitation, Applicants submit that this provides at least an additional reason why claim 5 defines subject matter that is patentable over UIML.

Examiner's response:

In response to Applicant's argument, the response above with respect to claim 1 is incorporated UIML disclose an XML language that permits a declarative description of a user interface in a highly device-independent manner. An objective of UIML is to permit a UIML document to be mapped to any type of user interface, from graphical to speech, and even to those not yet invented (see abstract). UIML clearly teaches the limitation receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device, See FIG. 1 which shows displaying data on multiple devices by receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device. Further, UIML does disclose presentation pattern since it's displaying data on multiple devices of it's (device) own natural language (see page 5, section UIML as a meta language) and furthermore, UIML disclose presentation pattern and further it is displays on a display device such as PDA or PC (see FIG. 1 and section UIML-Main Element). Applicant only makes general allegations. Therefore UIML does disclose the limitations as claimed and the rejection is maintained herein.

In the remarks, the applicant has argued that:

Claim 9, as amended, is similarly patentable over UIML because UIML does not disclose or suggest all of the limitations recited in claim 9. For example, UIML does not disclose or suggest a method that includes "customizing a workbench component that identifies constraints on the validity of the application specification document." The Examiner contended, in rejecting claim 9, that figure 1, figure 3, and related discussion of UIML disclose this aspect of claim 9. (Office Action, page 12). This is not correct. Neither figure 1 nor figure 3 of UIML show a workbench component or anything resembling a workbench component, and no mention or suggestion is made in UIML of identifying constraints on the validity of the application specification document. Furthermore, amended claim 9 recites "rendering a first object and a second object on the user interface of the device using the user interface model according to one of the layout themes for the device." UIML also fails to disclose or suggest this aspect of Applicants' claim 9. For at least these reasons, claim 9 defines subject matter that is patentable over UIML, as do each of dependent claims 10-14.

Examiner's response:

In response to Applicant's argument, the response above with respect to claim 1 and 5 is incorporated UIML disclose an XML language that permits a declarative description of a user interface in a highly device-independent manner. An objective of UIML is to permit a UIML document to be mapped to any type of user interface, from graphical to speech, and even to those not yet invented (see abstract). UIML clearly teaches the limitation receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device. See FIG. 1 which shows displaying data on multiple devices by receiving a document having a statement with an indication to render first and second objects and interpreting the statement to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device. Further, UIML does disclose presentation pattern since it's displaying data on multiple devices of it's (device) own natural language (see page 5, section UIML as a meta language) and furthermore, FIG. 1 of UIML clearly shows customized

deployment of UIML which is done on different type of devices and multiple types of language. Applicant only makes general allegations. Therefore UIML does disclose the limitations as claimed and the rejection is maintained herein

Specification

6. The objection to use of trademarks is withdrawn in view of Applicant's amendment/comments.

Claim Rejections - 35 USC § 112

7. The rejections under 35 U.S.C. §112 second paragraph to claims 11-14 is withdrawn in view of Applicant's amendment/comments.

Claim Rejections - 35 USC § 101

8. The rejection under 35 U.S.C. §101 to claims 8-14 is withdrawn in view of Applicant's amendment.

Double Patenting

9. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d

2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claims 1, 2 and 4-6 provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 1, 2, 3, 6, and 7 of copending Application No. 10/646,428 (hereinafter '428). Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following observation.

<i>Instant Claim</i>	<i>'428 Claim</i>
1. A method for rendering an assembly of a first object and a second object on a user-interface of a device, the device being either of a first type or of a second type, the first and second objects presenting data of an application, the method comprising the following steps: receiving an application specification document by the device, the application specification document having a statement with an indication to render the first and second objects in the assembly;	1. A method for rendering an assembly of a first object and a second object on a user-interface of a device, the device being either of a first type or of a second type, the first and second objects presenting data of an application, the method comprising: providing an interpreter specific for an application specification language used to write the application; storing the interpreter in the device; receiving an application specification document by the device, the application

<p>interpreting the statement of the application specification document to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device; and rendering the assembly of the first and second objects on the user-interface according to the presentation pattern identified in the interpreting step.</p>	<p>specification document having a statement with an indication to render the first and second objects in the assembly; interpreting the statement of the application specification document using the interpreter to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device; and rendering the assembly of the first and second objects on the user-interface according to the presentation pattern identified during the interpreting of the statement.</p>
<p>2. The method of claim 1, prior to the receiving step, further comprising: specifying the application in the application specification document by a workbench in a development computer; and simulating the rendering step by a pre-viewer component of the workbench.</p>	<p>2. The method of claim 1, further comprising: simulating the rendering of the assembly by a pre-viewer component of a workbench used in a development computer.</p>
<p>3. The method of claim 1, wherein in the rendering step, the first object and the second objects are rendered according to the</p>	<p>4. The method of claim 1, wherein in the rendering step, the first object and the second objects are rendered according to the</p>

<p>presentation pattern and to a predefined hierarchy pattern.</p>	<p>presentation pattern and to a predefined hierarchy pattern.</p>
<p>6. The method of claim 1, wherein the presentation pattern is as a display pattern, wherein the objects are rendered to the user-interface being a screen, and wherein the presentation pattern is identified according to the size (X) of the screen.</p>	<p>5. The method of claim 1, wherein the presentation pattern is as a display pattern, wherein the objects are rendered to the user-interface being a screen, and wherein the presentation pattern is identified according to the size of the screen.</p>
<p>7. The method of claim 1, wherein in the rendering step, the presentation pattern is an audio pattern.</p>	<p>6. The method of claim 1, wherein in the rendering step, the presentation pattern is an audio pattern.</p>
<p>8. A computer-program product to visually render a first object and a second object in an assembly on screen of a computing device, the objects presenting data of an application on a computer that is at least temporarily coupled to the computing device, the device being either of a first type or of a second type, the computer-program product having instructions that cause a processor of a computing device to perform the following steps: receiving an application specification document from the</p>	<p>8. A computer-program product to visually render a first object and a second object in an assembly on screen of a computing device, the objects presenting data of an application on a computer that is at least temporarily coupled to the computing device, the device being either of a first type or of a second type, the computer-program product having instructions that cause a processor of a computing device to: provide an interpreter specific for an application specification language used to</p>

computer, the application specification document having a statement with an indication to render the first and second objects in the assembly; interpreting the statement of the application specification document to identify a visual presentation pattern for the assembly from predefined first and second visual presentation patterns according to the type of the device; and rendering the assembly of the first and second objects on the screen according to the visual presentation pattern identified in the interpreting step.	write the application; store the interpreter in the computing device; receive an application specification document from the computer, the application specification document having a statement with an indication to render the first and second objects in the assembly; interpret the statement of the application specification document using the interpreter to identify a visual presentation pattern for the assembly from predefined first and second visual presentation patterns according to the type of the device; and render the assembly of the first and second objects on the screen according to the visual presentation pattern identified in the interpreting step.
--	---

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 102

10. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless --

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

11. Claims 1-14 are rejected under 35 U.S.C. 102(b) as being anticipated by the published document “UIML: An XML Language for Building Device-Independent User Interfaces” by Marc Abrams and Contanrinos Phanouriou (hereinafter, UIML) in December 1999.

Per claim 1:

UIML discloses:

providing an interpreter specific for an application specification language used to write the application (page 11, Figure 3 and related discussion, ...dynamic interface generate on-the-fly); storing the interpreter in the device (page 11, Figure 3 and related discussion, ...database); receiving an application specification document by the device, the application specification document having a statement with an indication to render the first and second objects in the assembly (page 3, section Deploying UIML “...UIML document...compiled to a target platform’s language...is mandatory for devices...cellular...” and page 5 section UIML as a meta-language “...UIML document...specifies a mapping of those names to a vocabulary specific to a particular target platform...” Also, see Figure 3); interpreting the statement of the application specification document using the interpreter to identify a presentation pattern for the assembly from predefined first and second presentation patterns according to the type of the device (page 5 section UIML as a meta-language “...UIML document...specifies a mapping of those names to a vocabulary specific to a particular target platform...” Also, see Figure 3); and

rendering the assembly of the first and second objects on the user-interface according to the presentation pattern identified during the interpreting of the statement (page 3, section Deploying UIML “...Java interpretive renderer permits the entire UIML interface to appear as a Java bean...end-user devices...” Also, see Figure 3).

Per claim 2:

The rejection of claim 1 is incorporated and further, UIML discloses: simulating the rendering step by a pre-viewer component of the workbench (See Figure 1 and 3 and related discussion).

Per claim 3:

The rejection of claim 1 is incorporated and further, UIML discloses: storing the predefined presentation patterns by the interpreter (page 11, Figure 3 and related discussion, ...database).

Per claim 4:

The rejection of claim 1 is incorporated and further, UIML discloses: wherein in the rendering step, the first object and the second objects are rendered according to the presentation pattern and to a predefined hierarchy pattern (See Figure 2a-2c and related discussion).

Per claim 5:

The rejection of claim 1 is incorporated and further, UIML discloses:

wherein the presentation pattern is as a display pattern, wherein the objects are rendered to the user-interface being a screen, and wherein the presentation pattern is identified according to the size of the screen (See Figure 1 and 3 and related discussion).

Per claim 6:

The rejection of claim 1 is incorporated and further, UIML discloses:

wherein in the rendering step, the presentation pattern is an audio pattern (page 2, section VoiceXML “...VoiceXML is a markup language for specifying interactive voice response applications...conversions...”).

Claim 8 is the computer product claims corresponding to method claim 1, and rejected under the same rational set forth in connection with the rejection of claim 1, above.

Per claim 9:

UIML discloses:

defining a user-interface model (page 4, section UIML-Main Elements “...user interface...set of interface parts comprising the interface...”);

defining an application specification document by a meta-language (page 5, section UIML as a meta-language “...UIML can be viewed as a meta- or extensible language...”);

customizing a workbench component that identifies constraints on the validity of the application specification document (See Figure 1 and 3 and related discussion);

defining layout themes for the computing device (page 5, section UIML-Main Elements "...a style element, which specifies presentation style that is device-specific for each class of interface parts...");

realizing the user-interface model in an interpreter component (page 4, section UIML-Main Elements "...user interface...set of interface parts comprising the interface..."); and

realizing the layout-themes in the interpreter component (page 5, section UIML-Main Elements "...a style element, which specifies presentation style that is device-specific for each class of interface parts...")

rendering a first object and a second object on the user interface of the device using the user interface model according to one of the layout themes for the device after receiving

(page 3, section Deploying UIML "...Java interpretive renderer permits the entire UIML interface to appear as a Java bean...end-user devices..." Also, see Figure 3), at the device, the application specification document, wherein the application specification document includes a statement with an indication to render the first and second objects in the assembly (page 3, section Deploying UIML "...UIML document...compiled to a target platform's language...is mandatory for devices...cellular..." and page 5 section UIML as a meta-language "...UIML document...specifies a mapping of those names to a vocabulary specific to a particular target platform..." Also, see Figure 3).

Per claim 10:

The rejection of claim 9 is incorporated and further, UIML discloses:

determining the types of tiles and the functionality of tiles, the tiles being elements of the user-interface model (page 5, section UIML-Main Elements “...style elements...mapping of interface parts to a vocabulary of names of user interface widgets in the platform to which the user interface will be mapped ”); determining relationships between the tiles in an assembly; and determining a navigation state and the required user operations on the navigation state (page 5, section UIML-Main Elements “UIML includes a peers element, which specifies what widgets in the target platform and what methods or functions in scripts, programs, or objects in the application logic are associated with the user interface”).

Per claim 11:

The rejection of claim 10 is incorporated and further, UIML discloses: defining specifications to the types of tiles (page 11, Figure 3 and related discussion, ...dynamic interface generate on-the-fly); defining attributes to express properties of the tiles; and defining attributes in the navigation state UIML uses three levels of names for interface parts and events. (page 6, section UIML as a meta-language “second name is in the style element and maps the mnemonic to an abstract widget name (e.g., MenuItem)... mapping from one abstract set of names (e.g., "BigWindow") to multiple platforms (e.g., MFC or Java Swing) without modifying the rest of the interface description... ”).

Per claim 12:

The rejection of claim 11 is incorporated and further, UIML discloses:

defining a representation on the output media of device for each element of the user-interface model; and defining the user-interface model for each operation of the user-interface model (page 5, section UIML-Main Elements “...style elements...mapping of interface parts to a vocabulary of names of user interface widgets in the platform to which the user interface will be mapped ”).

Per claim 13:

The rejection of claim 12 is incorporated and further, UIML discloses: creating models to specify the tiles and the assembly; implementing constructors to create user-interface instances from the application specification document; and implementing the user-interface instances from the models in a computer programming language (page 3, section Deploying UIML “...UIML document...compiled to a target platform’s language...is mandatory for devices...cellular...” and page 5 section UIML as a meta-language “...UIML document...specifies a mapping of those names to a vocabulary specific to a particular target platform...” Also, see Figure 3).

Per claim 14:

The rejection of claim 13 is incorporated and further, UIML discloses: implementing each layout-theme as a layout handler; and obtaining a selection of the layout-theme by a developer and forwarding the selection to the interpreter component (page 7, section UIML Architecture “The structure element enumerates a set of interface parts and their organization for various platforms and devices. The style element defines the values of various

properties associated with interface parts. The content element associates words, sounds, and images with interface parts to facilitate internationalization or customization for different user groups. Finally, the behavior element enumerates a set of rules that describe how the user interface should react on different stimulus (i.e., from user, device, or the application logic)").

Conclusion

12. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Satish S. Rampuria** whose telephone number is **(571) 272-3732**. The examiner can normally be reached on **8:30 am to 5:00 pm** Monday to Friday except every other Friday and federal holidays. Any inquiry of a general nature or relating to the status of this application should be directed to the **TC 2100 Group receptionist: 571-272-2100**.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, **Wei Y. Zhen** can be reached on **(571) 272-3708**. The fax phone number for the organization where this application or proceeding is assigned is **571-273-8300**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Satish S. Rampuria
Patent Examiner/Software Engineer
Art Unit 2191



WEI ZHEN
SUPERVISORY PATENT EXAMINER